Git and Github:

What is Git?

Git is a version control system. Think of it as a way to:

- Track changes in your code (or any files).
- Collaborate with others without overwriting their work.
- Restore previous versions if something breaks.

What is GitHub?

GitHub is a web-based platform that uses Git. It hosts your code online and allows collaboration, sharing, and management of Git repositories (repos).

Basic Workflow of Git

Here's a high-level view of how Git works:

1. Initialize a project with Git so you can track changes.
2. Make changes to your files.
3. Stage the changes (tell Git what you want to track).
4. Commit the changes (save them into the version history).
5. Push the changes to a remote server like GitHub (optional but important for collaboration).

Step 1: Installing Git

Before you can use Git, you'll need to install it:

- Windows: Download and install Git from git-scm.com.
- Mac: Use Homebrew: `brew install git`.
- Linux: Install via the package manager: `sudo apt-get install git`.

Step 2: Configuring Git

After installing Git, you need to set your name and email (this is how your commits will be labeled):

```
git config --global user.name "Your Name"
git config --global user.email "you@example.com"
git config --list
```

Starting with Github Commands:

ls-To list all files
Ls -a or ls -Force- to show hidden files.

Git init-to initialize any vs code folder as git folder or repo.
Git add . /git add file_name
Git commit -m”initial commit”
Git remote add origin repo_link
Git push -u origin main

Git remote -v: to know which repo are you in
Git branch -m main:to change branch

Cloning a Remote repo:
Git clone repo_link
Make changes
Git push origin main

Forking a Remote Repo:
Click on Fork
Git clone repo_link
Make changes
Git push origin main
Make a pull request.

Branching:
Give example of what’s app like what’s app was full fledged but adding communities ,channels,payment option were additional features or branches you may say.

Creating a Branch

git branch new-feature

This command creates a new branch named new-feature. This branch is an exact copy of the branch you're currently on (usually main).

Switching to a Branch

```
git checkout new-feature
```

This command switches your working directory to the new-feature branch. Any changes you make now will only affect this branch.

Or, you can use:

```
git switch new-feature
```

Listing Branches

```
git branch
```

Merging Branches Once you've finished working on your branch, you can merge it back into the main branch:

```
git checkout main
git merge new-feature
```

Delete the new-feature branch:

```
git branch -d new-feature
```